Research report 183

EXTENDING CASSANDRA

IAIN D. CRAIG

(RR183)

In this paper, I record a number of extensions to my CASSANDRA architecture. Many of these extensions formed part of the original conception. Since the original architecture definition was published, these extensions have become more pressing. In particular, I argue that the speech act model can be used to great effect by DAI systems, and I also indicate the variety of ways in which reflective reasoning can be used. The remarks on reflection are founded on work that I have already done on this topic. The extensions form the next stage of the development of the CASSANDRA architecture

Department of Computer Science
University of Warwick
Coventry CV4 7AL
United Kingdom

March, 1991

1 INTRODUCTION

In this note, I want to document a number of thoughts I have had on the extension of my CASSANDRA architecture (Craig, 1989). The document is intended to record my thoughts for future reference: it seems opportune to record these thoughts now. In order to describe the extensions, it will be necessary to record the development of CASSANDRA: this puts things into a proper perspective.

The extensions I have in mind concern the inferential capabilities of Level Managers. More specifically, they consist of the introduction of meta-level structures (including a more refined control structure), a declarative database and a richer concept of communication. In addition, given its historical roots in the intelligent agent concept, another class of extension deals with *dynamic* level managers and *dynamically-developing* communications paths through a network of Level Managers (they are always static in the 1989 definition). All of these extensions relate, quite definitely, to the role of CASSANDRA as a distributed systems architecture, although the dynamic aspects will be ingored for the most part below.

As I hope will be seen, these extensions represent a genuine development of the architecture—sometimes, indeed, they subvert the original conception.

For the remainder of this note, I will refer to the definition of CASSANDRA that appears in the published book (Craig, 1989) and whose Z specification is to be found in the new book (Craig, 1991a) as CASSANDRA89. Furthermore, because of the newer perspective on the architecture, what is called the Level Manager construct in CASSANDRA89 will now be more simply termed an *agent*.

2 ANCIENT HISTORY

At the time CASSANDRA89 was first defined (during 1985), I had in mind the idea of a problem-solving system as being composed of many thousands of autonomous agents, which communicate when attempting to solve problems. This idea is neither new nor novel (see, for example, Hewitt's ACTORS (1977), or Minsky's (1985) *Society of Mind* theory.

In my conception, agents could come into being and be destroyed: they had lifetimes. During the life of an agent, it attempted to solve a part of a problem (this aspect is similar to that of an ACTOR). Indeed, a single agents could participate in the solution of many problems—what each problem had to have in common was that it required the knowledge provided by the agent.

At the time of its original conception, the Agent Model seemed to be impossible. With time, it became clear that each agent should be rather larger and be capable of rather more than I had first thought. The original CASSANDRA system was a first attempt at this: it was inspired by the blackboard model of problem solving, and the first implementation was based on the idea that each Level Manager should roughly correspond to an abstraction level in the older architecture. The idea that Level Managers correspond to abstraction levels is very much in evidence in the published book (Craig, 1989).

In the years that have passed since the original development, my ideas on CASSANDRA and related systems have changed somewhat. It has become clear that the size of agents is quite considerable—this is because there is a lot they have to do. In addition, the idea of a *symbiotic system* has begun to appeal to me. In systems of this type, the human user is part of the system. The user participates in the development of solutions to problems, and what used to be Level Managers have to communicate with the user in relatively sophisticated ways.

I have become somewhat dissatisfied with a number of aspects of CASSANDRA89.

It was always part of the architecture that inter-agent communications should become more sophisticated—this is witnessed by a number of remarks in Craig, 1989, in particular the reference to speech acts (Austin, 1962; Searle, 1969). Also, I was always dissatisfied with control in CASSANDRA89: with time, I have wanted to get away from the strictly blackboard approach to control and have wanted to develop more powerful (and, in a certain sense, more reasonable and reasoned) control mechanisms. Finally, it always seemed to be a flaw in the original conception of the system that Level Managers could not manipulate and reason about representations of themselves and their relationships to other Level Managers in a system (and to the user in a symbiotic system).

Again, it must be noted that these requirements, ideas and faults are not particular to CASSANDRA89: they apply to other DAI systems, as well. When I first defined the architecture, some of these ideas were clear in my mind—others have become clearer with time. Other workers have made similar comments about DAI systems; indeed, some have suggested speech acts as a communications protocol, and others have remarked that agents need to reason about themselves and their position in a DAI system. What I think is new is the extension of CASSANDRA89 to cope with these additional requirements.

Finally, and to close this section, it needs to be noted that the original view of CASSANDRA I had was of a community of agents. The emphasis is definitely on community. Part of the motivation for this concept was the observation that most AI systems operate in a 'solipsistic universe': that is, one in which the AI program is the only agent. In such a universe, there is no need for the AI program to act or even to take action into account, and it need not reason about itself (for there is no point).

During the initial development, it was pointed out to me[1] that people do not operate in a vacuum, and the information resources are available *by virtue* of the fact that they live in a world populated by other people and other objects (a concept which now seems familiar from the concept of situated action (Suchman, 1987)).

3 COMMUNICATIONS AND SPEECH ACTS

In the book (Craig, 1989), I suggested that the speech act theory of Austin (Austin, 1962; Searle, 1969) might be a good basis for describing the communications aspects of CASSANDRA89. Communications in the reported CASSANDRA systems was, perhaps, the weakest aspect, and was also the closest to traditional distributed systems: the reason for this is that the book reports on work that was undertaken to show the viability of the CASSANDRA approach, in particular the viability of Level Managers.

Since the time the work reported in the book was undertaken, the speech act theory has appeared to be increasingly attractive. The fundamental appeal of the speech act theory should be apparent from the accounts given by Austin (*op. cit.*) and, in a later and more elaborate formulation, by Searle (*op.cit.*), although a number of problems for describing natural languages with this theory are examined by Levinson (1983)[2].

Although the concept of a speech act should be familiar to AI workers (primarily from the work of Cohen and Perrault (1987), it is worth spending a short while in justifying them as a basis for an approach to communications in a DAI system. The reader should note that we disregard the distinction between illocutionary and perlocutionary acts in the discussion: this is for simplicity of presentation.

The most basic appeal of the speech act theory is that it makes explicit the intentions that lead to an utterance. Indeed, part of the justification of a speech act is that, by performing a speech act, a speaker makes clear his or her intention to the hearer. For example, in saying "I promise to send you the tape", a speaker is making clear the fact that, at some time in the future, the speaker will indeed send the tape to the hearer. The hearer, by hearing this utterance, can infer (assuming that the speaker is truthful) that the speaker has the intention of sending the tape. The hearer is also able to infer other things about the speaker, quite apart that he or she will send the tape. For example, the hearer can infer that the speaker is in or can gain possession of the tape, and that the speaker also has the means to send the tape to the hearer.

Searle's (1969) book contains an attempt to account of the conditions under which a speech act is successful. In other words, he attempts to account for the preconditions

[1]By Dr. John Hughes, Department of Sociology, University of Lancaster, and I now publicly thank him for this observation.

[2]The objections outlined by Levinson deal with *indirect* speech acts—in an AI system, it may well be possible to avoid indirect speech acts.

which enable a speech act to be performed and the conditions under which, when they are satisfied, the speech act can succeed. As a consequence, the theory presents a set of conditions which can be inferred by a hearer when understanding an utterance. This fact is important because part of the speech act theory is that it determines a structure for communication—this amounts to more than a theory of meaning. For a speech act to be successful, both parties must understand the rules which apply to performing an act of communication (cf. Grice, 1957).

A number of properties make speech acts an extremely attractive framework for communications in a DAI system. In particular, there is the property that the rules which are used in the production of utterances can also be used in their interpretation. These rules, some of which are explicitly given by Searle (1969), relate the speaker's intentions and utterances: speech acts make the connection clear. This explicit connection can be exploited by agents in a DAI system when making inferences about the state and intentions of other agents.

A significant problem in a DAI system is that of inferring the state and intentions of other agents in the system. The problem reduces to that of building a model of what is happening inside other agents, as well of as forming a global picture of the state of the system as a whole. This model is then used by an agent to determine what to do next. For example, models of the external environment are required by an agent when it needs to select a sub-problem on which to work. Quite clearly, it rarely makes sense for the agent to duplicate work that it being done elsewhere in the system[3]. It equally makes no sense for an agent to concentrate on some sequence of actions which do not fit into the solution that is emerging elsewhere in the system (i.e., agents should not engage in divergent behaviours).

Given the fact that the communications medium in a distrubuted system can only ever have a finite bandwidth, it is not possible for total state information to be passed between agents. Furthermore, if total state information were passed across the communications medium, agents would have to engage in a costly analysis exercise in order to obtain information that is both relevant and useful to them.

These facts have the consequence that agents *must* infer the state of other agents and of the whole system from a limited amount of information.

In addition to the state problem, there is the task-allocation problem. In the published accounts of CASSANDRA, task allocation was performed on an *a priori* basis—each Level Manager was assigned a task or set of tasks that it had to perform. In the original (unpublished) conception of CASSANDRA, task allocation was a very much more dynamic process (in some ways reminiscent of the Contract Net protocol (Davis and

[3]Sometimes, duplication might be warranted—for example when the a system is engaged in the competitive formation of solutions.

Smith, 1983)). Dynamic task allocation can only be performed when agents are aware of what other agents are doing, what their state is, and what they are capable of. In a dynamic environment, the transmission of this information requires using more of the available bandwidth. In the course of general problem solving, it may be necessary for an agent to try to get other agents to take over sub-tasks (unless each agent has enough power to solve all the problems that might be assigned to it): in other words, task allocation may, under some circumstances, turn out to be a normal operating problem for DAI systems of certain types (in particular, the type of system I originally conceived CASSANDRA to be).

One solution (proposed by Durfee and Lesser, 1987) is that agents exchange plans of their actions: these plans (called 'partial plans') are meta-level structures which represent partial information about the state of the agent. In the context of human interactions, the sources of information are richer. In particular, people send each other messages and hold conversations. Sometimes, these conversations are of a nature that can be accounted for under the general heading of plans, but the exchange of plans entails the exchange of relatively large amounts of information. The partial plans model entails the exchange of information between agents, and, in the limit, entails the exchange of relatively large amounts of information. This information, as has been noted, is used by agents to construct a model of the total state of the system during problem solving.

The speech act model involves the exchange of rather less information, even though it does require that agents engage in rather more inference. In short, the speech act account requires that agents analyse the communications which they receive from other agents; they must also produce communications as the result of an inferential process (an agent has, on the basis of its state, to infer that it needs to engage in communication). Because the production and understanding of communications proceeds on the same (or closely related) bases, an agent can analyse the communication and hypothesise reasons why it was performed. In other words, because of the rule-governed (or conventional) nature of speech acts, analysis of speech acts can lead to the discovery of information about other agents and their internal states (for example, if I say "Please, pass the butter", a hearer is entitled to infer that I do not have the butter and that I would like some)—assuming, as Searle does, that each act of communication is undertaken in good faith (the assumption that *speakers tell the truth*, in other words).

In CASSANDRA89, all communications were point-to-point in the sense that broadcast and pooled communications were not possible. In the Z specification (Craig, 1991a), I made explicit provision for message store-and-forward and broadcast by allowing routing information to be used[4]. Store-and-forward is a relatively low-level process and does not institutionalise communications. However, there is no restriction in the architecture definition itself that one Level Manager may communicate *only* with other Level Managers that are working on similar or related problems: that restriction seems to be a reasonable one in an implementation, but is not enforced by the definition proper (and is certainly not forbidden by the Z specification).

Within the context of CASSANDRA89, communications are, therefore, limited in scope, although it is possible for one CASSANDRA89 Level Manager to ask another for information about a third. (Given the fact that my original vision of CASSANDRA had Level Managers directly communicating with many—perhaps tens—of others, the amount of information that can be derived from direct communications appears to be quite large.) A further restriction that was enshrined as an architectural principle in CASSANDRA89 was that communication channels were established when the system was configured. In the version of the architecture which is currently being developed, communication paths are much more dynamic: channels can be opened at any time during a run of the system—all that an agent needs to know is the name of the agent with which it wants to communicate (perhaps *descriptions* could be used, but it is not yet clear what form these descriptions would take).

Even in this relatively limited context (one that is closer to conversations between people), the speech act model has application. In particular, it is possible for models of other Level Managers to be constructed on the basis of their acts of communication. That is to say, it is not necessary for state information to be communicated as a matter of routine.

At this point, it is necessary to observe that I do not view a CASSANDRA system as requiring the full range of speech acts (Austin, for example, claimed that he and his students had identified well over one thousand verbs that acted as speech act indicators). An initial set might be: asking, telling, ordering, requesting, denying, asserting, promising, instructing (in the didactic sense, also), doubting, informing. Such a restriction, which might need some augmentation, but, I hope, not much, clearly aids in the development of a system and also helps in ensuring that the system performs with less delay than would otherwise be the case.

So far, it has been argued that the speech act model can be used as the basis of communication production and understanding[5]. Cohen and Perrault (1987) have

---

[4]It seems reasonable, in some circumstances, to allow communication structures such as pooling (a concept similar to bulletin boards), or to allow distribution or mailing lists to be maintained. Any extensions to the communications structure at this level would, though, appear to be related to the organisational model being employed (see next section).

[5]It is important to recognise the difference between a communication and a message. In our terms, a message consists of the exchange of one block of information (the idea is that a block or message is something like a single packet), whereas a communication can involve the transmission and reception of a sequence of blocks—clearly, a communication may involve the exchange of only one block. What is essential, though,

proposed a plan-based model of speech acts for language production. The Cohen and Perrault theory could be used as a basis for one half of what is needed in support of the above. The other half requires a theory of speech act recognition and goal inference. This does not appear to be as demanding as might appear: the explicit nature of speech acts which has been mentioned a number of times seems to come to the rescue. The basics of a recognition model appear to be relatively straightforward: what appears to be more difficult is the integration of the information derived from the recognition process with the state of the agent which performs the recognition. This last aspect remains a topic for further investigation, although it is clear that it presupposes that each agent possesses knowledge of other agents.

The speech act model does not rely entirely upon the illocutionary and perlocutionary force of the communication itself. The determination of the conditions under which a speech act can be performed can be seen as a kind of context-independent component of the theory. In addition, each communication contains information—this has been termed the propositional content of the speech act. Clearly, the propositional content is context-dependent; it is also clear that the speech act, often in a clear fashion, conditions the manner in which the propositional content is treated. In a DAI system, the propositional content determines what the message contains. In other words, the content of a communication should not be overlooked.

Finally, whenever two agents communicate it is true that the communication has been effected. In other words, *the very fact* that there has been a communication is an item of very relevant information. The importance of this information varies with context: when dealing with a loquacious or voluable person, verbal communication tends to lose its value, but when an otherwise taciturn person says something, we are entitled to place some emphasis on the fact that they have spoken. At the moment, I am not sure how this additional information fits into the overall approach, although I am convinced that it has a role to play.

To summarise this section. Early in the development of CASSANDRA89, the speech act theory presented itself as a way of regulating inter-agent communciations. With time, the importance of the theory within the CASSANDRA model has increased, and richer properties of communication have been observed. Clearly, communications is not the entire story in DAI, but a rich model that gets away from the traditional computer science view is clearly necessary, particularly when the range and flexibility of agent behaviours is increased.

4 ORGANISATION

---

is that for any communication, there is an identifiable speech act. A communication is a meaningful unit of exchange.

The concept of organisation was also clear during the work on CASSANDRA89. The way in which the Level Managers of that version were organised appeared to be a major factor in the construction of adequate systems; it is also crucial in the development of adequate theories of group problem solving. At the time of writing, organisational models have not been developed, although the ways in which organisation impacts upon the operation of agents has become clearer. In the CASSANDRA model, organisation is considered in two lights:

(i) It represents a way of organising knowledge and capabilities in a system.

(ii) It represents a structure that is to be imposed on the communications that are possible between agents.

I view these points as being of equal importance.

The first of these points can be explained as follows. An organisation determines the kinds of agents that will be present in a system. Each kind of agent will be equipped with certain capabilities and certain knowledge: the organisational role determines what the agent will be capable of. For example, a project manager needs managerial skills and an overview of his (or her) project; a programmer, on the other hand, needs to be able to program and only needs to concentrate on one small aspect of the project; project managers do not usually engage in programming and programmers do not usually undertake managerial (say, scheduling) tasks.

The types of communication that can be expected are also determined by the organisation. Managers tend to send messages that contain high-level information more general in scope than programmers; they also tend to send directives and requests. A programmer working under a manager does not, in general, send the manager demands, nor do workers tend to issue instructions to superiors—they may send requests of a more or less specific nature. It is in this general sense that the organisation determines the kinds of communications that are initiated by its agents.

The above leads, quite naturally, to the observation that an organisation entails *expectations* about the kinds of communications that will be undertaken by various agents in an organisation. When an agent engages in a kind of communication that is not predicted by the organisation, it is reasonable to assume that the agent's role in the organisation has changed. The role an agent plays in an organisation also, equally naturally, determines the kinds of communications that it may initiate and expect to participate in. In other words, once an organisational structure is in place, agents can determine the ways in which they are expected to behave and can also make predictions about the behaviour of other agents.

Organisation remains an area in which the CASSANDRA model needs to be extended: it has not, so far, been the focus of much attention.

5 SELF-REASONING

Very early in the development of CASSANDRA, it became clear that agents (then called Level Managers) should be able to reason about themselves in a variety of different ways. This was coupled with the aim of providing agents with an extensive and powerful meta-level. Self-reasoning, or reflection, has been the focus of more recent efforts. The more recent work has resulted in a better understanding of the ways in which reflection fits in to the general picture of CASSANDRA that I developed in the mid 1980s.

In this section, I will explain the reasons why autonomous agents must reason about themselves, and will explain some of the kinds of reflective inference that I see as important: it might well be the case that the list has to be extended in the light of future experience. For this section, it is necessary to keep in mind the idea that agents are autonomous.

In a DAI system, an agent needs to know its strengths and weaknesses. It needs to know which problems it can solve, which problems it can participate in (and how), and which problems it can neither produce a complete solution nor participate in a solution. It needs to know how to make the best use of what it knows and what it can do, and this includes strategies for coping with its weaknesses. These points apply, of course, to any knowledge-based system, but they apply with increased force in a distributed environment, particularly one which is dynamic in nature.

The environment within which a system operates may be more or less dynamic. In more dynamic environments, it is necessary for each agent to be as flexible as possible in its responses to other agents and to the external (non-agentive[6]) environment. The increased flexibility of DAI agents is crucial because the external environment assumes a far greater importance than in centralised systems, and because the external environment can change very rapidly (indeed, the environment generated by the system may itself undergo rapid change). The external environment, for many systems, provides the data upon which the system operates. If we extend the terminology to the environment which is external to an agent (which contains other agents as well as the external environment or outside world), the crucial role of the environment becomes even clearer: as has been stated (section 3 above), agents only have a partial view of the state of the overall system—non-local changes can occur asynchronously and have to

[6]The term *external environment* is intended to refer to that part of the environment in which the system operates which is not composed of the agents which form the system. The external environment may contain other computational structures, it may contain people, it may contain other systems of different kinds (another DAI system, for example), or it may be composed of none of these.

be detected in some way if an agent is to form a picture of what is happening within the system.

It seems to me that the environment in which an agent finds itself is a significant reason for wanting agents to be as flexible as possible.

A further property of autonomous agents is that they must be able to reason about their relationship to other agents and to the environment. Part of this is to do with the formation of models of the system state. Organisational information, as well as communication, play a part in determining how an agent relates to everything else. In other words, agents must reason about the communications they receive (see section 3) and they must reason about their own role in the current organisation, and about the role played by the agents with which they communicate. A simple illustration of this is the case of an agent which needs additional information: to obtain the information, the agent can engage in communication with other agents to determine where the necessary information can be obtained. In a hierarchical organisation, the agent might enquire of its immediate superior where to obtain the information, for example.

The role of reflection is also important in dynamic task allocation. Before an agent can undertake to perform a task (solve a problem), it must determine that:

(i) It has the capability—in terms of knowledge, time and processing power—to perform it (perhaps by delegating part of the task to other agents which it knows to be suitably equipped).

(ii) It has a method for performing the task.

The second point may be interpreted as saying that the agent has strategies available that will enable the solution of the problem. Alternatively, it may be interpreted as saying that the agent has a suitable interpreter. This last point relates to some of my more recent work on developing reflective structures which can undertake the interpretative task in more than one way (Craig, 1991b).

The relevance of this is as follows. By providing a system with different interpreter mechanisms, the system is thereby able to interpret information and apply knowledge in different ways. In the ELEKTRA system (Craig, 1991c), it is possible to reason about which rule interpreter to use in applying the rules in the system. This is made possible by the existence of a potentially infinite sequence of meta-levels, each of which reasons about lower levels. In ELEKTRA, the choice of rule interpreter is a knowledge-based activity.

This fact suggests that, in a CASSANDRA system, it should be possible to choose the way in which knowledge is applied as the result of an inferential process (this is quite apart from the application of strategic and tactical knowledge, it should be noted). It also suggests that additional flexibility can be obtained by allowing the system to reason about its own behaviour when expressed as the interpretation of knowledge-encoding structures.

There is another sense in which self-reasoning can be applied: this other sense relates to the way in which declarative knowledge is interpreted. One failing of CASSANDRA89 is that it made no provisions for a declarative database: everything was expressed in terms of Knowledge Sources. For a very long time, I have considered that a declarative component was necessary, and have tried (for example, Craig, 1987, for an example within the blackboard framework) to integrate declarative structures with the procedural representation afforded by Knowledge Sources; a more recent and extensive exploration is currently being undertaken (Craig, 1991d).

The more recent work takes its direction from the account given in Craig, 1991e. There, I concentrated on the lessons that can be learned from the role of the meta-language in first-order logic. In first-order systems, it is the meta-language which contains the machinery for determining truth and reference. I argued that these concepts are essential for AI systems in general, and for reflective systems in particular. This is because a reflective system needs to be able to give an account of truth in its system. In short, the meta-language gives an object-language its semantics. This entails that the meta-language determines the interpretation map for the object-language[7].

The major lesson is that the semantics of knowledge representations must be given explicitly. The main problem is asking what a given sentence of the representation language means. This is not merely a computational problem (in other words, it does not reduce to specifying an interpretive algorithm), but a logical one. For a reflective system such as those I am currently considering, the system itself must contain structures which determine the truth-values of representation language sentences.

For a DAI system, this problem is clearly of considerable importance because the system's agents have to reason about themselves and about their environment (taken as including other agents, communications, etc.): they need to believe only that which is true. This is shown quite clearly in the work by Konolige (1982, 1986, 1988), Moore (1985, 1988) and others on mutual knowledge and belief. The problem is also shown by the fact that communication must be 'meaningful' (in some sense): the mere exchange of uninterpreted symbols is just that—the exchange of *uninterpreted* symbols. What is required is some interpretational mechanism (in the logical sense) which will determine the meanings of the exchange.

Given the different kinds of knowledge that an intelligent, autonomous agent must possess, it is essential that the agent be able to use that knowledge in meaningful ways. It must be able to interpret processes and events over which it has no control just in

[7]The next question is 'what determines the interpretation for the meta-language?' The answer, according to Tarski, is that a meta-meta-language does.

order to react in appropriate ways. The variety of knowledge that an agent must possess entails that the agent must be able to use that knowledge in the most effective and flexible ways possible. Furthermore, it must be able to reason about what it knows, both in terms of scope and also in terms of meaning. This implies that extensive meta-level structures must be employed by the agent in reacting to its environment and in solving problems. To take an example, it must be possible for an agent to determine the likely effects of engaging in communication with another agent: this requires that it reason about the communication, about the reason for engaging in the communication, and that it reason about the agent with which it intends to communicate.

The area of self-reasoning is the focus of current research.

6 CONCLUSIONS

This paper has documented my thoughts on how to extend the CASSANDRA89 architecture. Some of the extensions were part of the original conception of the CASSANDRA architecture, whereas others have become clear since that time (for the most part, during 1988-90). The extensions which have been worked out in detail concern the role of communications and of reflection in CASSANDRA agents. The other main extension is the enrichment of the concept of organisation. Furthermore, there is an increasing need for semantics in DAI systems: because of the nature of the environment in which such systems operate, and because of the constraints imposed on component agents by being elements of such a system, it is becoming increasingly clear that they must be in a position to reason about their own representations and the ways in which they are used.

The reader will have noted that I have not bothered with the problems of parallel and distributed implementation. The reason for this is that I consider these problems to be of a merely implementational nature: in a sense, they pose no conceptual problems.

The CASSANDRA architecture is now being used by a number of other workers, mostly in the 1989 version. In the near future, I hope to produce a new version of the architecture which incorporates some, if not all, of the extensions that have been outlined above.

REFERENCES

Austin, J.L., *How to Do Things with Words*, OUP, 1962.

Cohen, P. and Perrault, R., Elements of a Plan-based Theory of Speech Acts, *Cognitive Science*, Vol. 3, pp. 177-212, 1987.

Craig, I.D., *BB-SR*, Research Report No. 94, Department of Computer Science, University of Warwick, 1987.

Craig, I.D., *The CASSANDRA Architecture*, Ellis Horwood, Chichester, England, 1989.

Craig, I.D., *The Formal Specification of AI Systems*, Ellis Horwood, Chichester, England, *in press*, 1991a.

Craig, I.D., *Rule Interpreters in ELEKTRA*, Research Report, Department of Computer Science, University of Warwick, *in prep.*, 1991b.

Craig, I.D., *ELEKTRA: A Reflective Production System*, Research Report, Department of Computer Science, University of Warwick, *in prep.*, 1991c.

Craig, I.D., *The Formal Specification of RPS*, Research Report, Department of Computer Science, University of Warwick, *in prep.*, 1991d.

Craig, I.D., *Meta-Knowledge and Introspection*, Research Report, Department of Computer Science, University of Warwick, *in prep.*, 1991e.

Davis, R. and Smith, R.G., Negociation as a Metaphor for Distributed Problem Solving, *Artificial Intelligence Journal*, Vol. 20, pp. 63-109, 1983.

Durfee, E.H. and Lesser, V.R., Using Partial Global Plans to Coordinate Distributed Problem Solvers, *Proc. IJCAI*, pp. 875-883, 1987.

Grice, H.P., Meaning, *Philosophical Review*, pp. 377-88, 1957.

Hewitt, C.E., Viewing Control Structures as Patterns of Passing Messages, *Artificial Intelligence Journal*, Vol. 8, pp. 323-364, 1977.

Konolige, K., A First-Order Formalisation of Knowledge and Action for a Multiagent Planning System, in Hayes, J.E., Michie, D. and Pao, Y. (*eds.*), *Machine Intelligence 10*, Ellis Horwood, Chichester, England, 1982.

Konolige, K., *A Deduction Model of Belief*, Pitman Research Notes in Artificial Intelligence, 1986.

Konolige, K., Reasoning by Introspection, in Maes, P., and Nardi, D., *Meta-Level Architectures and Reflection*, North-Holland, Amsterdam, 1988.

Levinson, S.C., *Pragmatics*, Cambridge Textbooks in Linguistics, CUP, 1983.

Minsky, M, *The Society of Mind*, Heinemann, London, 1985.

Moore, R.C., *A Formal Theory of Knowledge and Action*, Report No. CSLI-85-31, CSLI, Stanford University, 1985.

Moore, R.C., Autoepistemic Logic, in Smets, P. et al. (*eds.*), *Non-Standard Logics for Automated Reasoning*, pp. 105-136, Academic Press, London, 1988.

Searle, J.R., *Speech Acts*, CUP, 1969.

Suchman, L.A., *Plans and Situated Actions*, CUP, 1987.